THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 58, No. 10, December 1979
Printed in U.S.A.

An Interactive Terminal for the Design of Advertisements

By B. E. CASPERS and P. B. DENES

(Manuscript received July 9, 1979)

The advertisement below was created on the experimental graphics terminal described in this paper and demonstrates its features.

The terminal is intended for the interactive design of advertisements. The work is part of our research on replacing photographic by digital methods of picture handling in typesetting; the results may help reduce the cost of producing Yellow Pages directories. Controls of the terminal were planned with human engineering in mind and thus are easy to learn and use.

The experimental terminal consists of a color TV *display, a keyboard, a lightpen, a facsimile-type hard copier, and a fast picture scanner/digitizer which are all connected to a minicomputer equipped with disk and tape. It enables the user to interactively position, size, crop, and edit pictures as well as text; pictures can be scanned and digitized in a few seconds, corrected, and stored for future use; easy-to-use commands are provided for enhancing the appearance of text or pictures by automatically "outlining," "shadowing," or "screening" interactively selected areas. Finished advertisements are either stored on disk or outputted to tape; the data on the output tape are in a form ready for use on a* CRT *phototypesetter. The advertisement can also be outputted more quickly but with reduced quality using the hard copier.*

Ways of outputting graphics on the typesetter with high quality and speed have also been studied; an average output speed of 1.5 seconds per square inch was achieved.

Field trials of the terminal are now in progress. Their goal is to study the type of personnel best suited to operate the terminal, to establish easy-to-use operating procedures, to make time and motion comparisons with existing production techniques, and to identify modifications for improved operation. Results already available are discussed in this paper.

I. TYPESETTING TECHNIQUES—OLD AND NEW

Traditional typesetters used lead type to obtain an image of the page to be printed, while more modern machines use computer-controlled CRTs. Indeed, modern printing technology uses computers not only for typesetting but also for preparing, editing, storing, and paginating the material to be printed. So far, however, computers have been used for processing only the text to be printed, while pictures are handled photographically. Photographic preparation of the picture material includes sizing and cropping, as well as such enhancements as "screening" (simulated gray scale), outlining, and shadowing; finally, the photographs are pasted manually on the page, into spaces appropriately left blank by the text typesetting process. Such photographic handling is used extensively, it is labor-intensive, and therefore it is expensive.

II. OBJECTIVES OF OUR TYPESETTING RESEARCH

Our aim is to replace these *photographic* methods of picture handling by *digital* methods. This would result in strongly reduced involvement of manual labor in typesetting tasks such as Yellow Pages directory production: It would eliminate pasting down pictorial material, much of the photographic laboratory work, and the manual retrieval/storage of artwork and advertisements. Digital methods also open the way for interactive design of the material to be typeset: Pictures (and text) can be moved, sized, cropped, or enhanced in other ways, and the changes can be viewed immediately on a display screen.

The principal steps needed for digital handling of pictures are:

(*i*) Scanning/digitizing of the original art.

(*ii*) Coding of pictures for efficient storage.

(*iii*) A design terminal for laying out advertisements and for adding special effects such as screening.

(*iv*) Appropriate programming of the typesetter to produce the pictures on its output medium.

It should be noted that digital typesetting of pictures requires that the picture data be converted into one code for efficient storage, into another code for aesthetic processing such as sizing and outlining, and into yet another code for fast typesetting. Means for converting quickly from one code to another is therefore a key element of the entire process; it occupies much of our attention, and work on it continues.

Bell Laboratories research has already made significant progress in finding methods of coding digitized pictures for efficient storage.[1-3]

This paper describes our research on an interactive terminal and on certain aspects of finding the best code for controlling typesetters. We also describe how the design of the man-machine interface was changed to simplify the use of the terminal by staff not familiar with computer operations. We have conducted a field trial to test the usability of this type of terminal. A more detailed trial is now starting which includes time and motion study comparisons with conventional production methods, attempts to arrive at a better definition of personnel requirements, and studies of various operational enhancements.

III. THE INTERACTIVE TERMINAL

The first version of our advertisement design terminal was implemented on a Honeywell DDP 224 laboratory computer.[4,5] An improved version with many new features was transferred to a stand-alone minicomputer so that it could be installed for field trials at directory production departments of telephone companies. This is the experimental terminal described in this paper. The terminal performs the following functions:

(*i*) It allows text and picture material to be viewed and adjusted relative to each other; pictures and text can be moved, sized, and cropped, and text can be edited; the appearance of text or pictures can be enhanced by outlining, shadowing, and "screening" (shading).

(*ii*) The entire advertisement is coded suitably and then outputted to a digital electrostatic printer (200 lines/inch resolution).

(*iii*) The advertisement is also coded for output to a CRT typesetter (720 lines/inch resolution).

IV. THE HARDWARE

The principal hardware components used for the experimental terminal are shown in Fig. 1 and detailed below (of particular interest are the TV scan display and the scanner; reasons for their selection are given):

(*i*) A 16-bit word-length minicomputer (Data General Nova 830) with 48k words of memory, a tape drive, and three 2.4 Mbyte disks. The main purpose of the tape output is to transport advertisement definitions computed in typesetter code by the terminal to the typesetter for final outputting. The disks provide about 1.5 Mbyte buffer space for program execution and about 6 Mbytes for storing picture and advertisement specifications. A 96 Mbyte disk was later added to accommodate the larger number of logos and fonts needed for the extended trial, which is now starting.

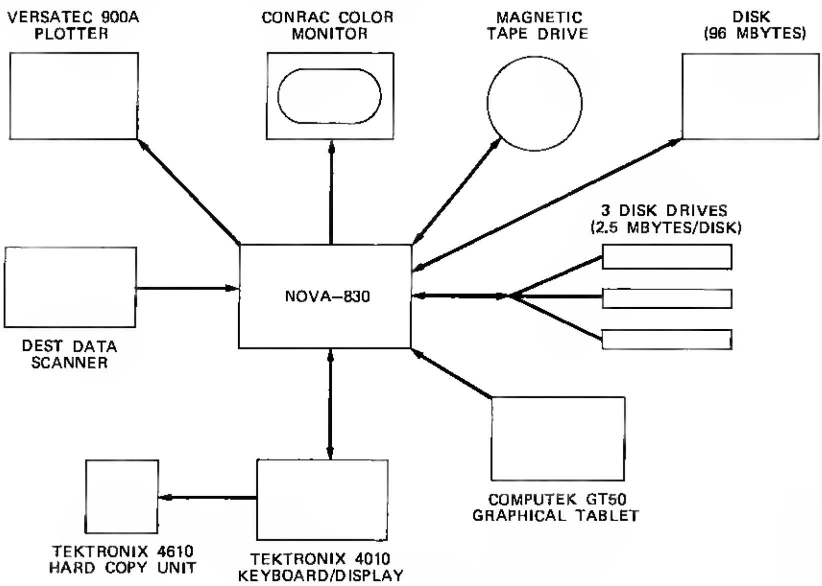

Fig. 1—Block diagram of terminal.

(*ii*) A keyboard/display (Tektronix 4010). Its main purpose is to provide the "function keys" by which the layout design process is controlled. The keyboard/display is also used for typing in text and for outputting messages to the operator.

(*iii*) A tablet/stylus (Computek) for easy communication with the displayed material by "pointing."

(*iv*) A color television monitor (Conrac) with buffer memory provides the main display on which the operator can inspect entire advertisements and individual scanned pictures. The entire display is defined in terms of $512 \times 508 = 260096$ picture elements; each picture element (pixel) is specified by two bits which allow selection of one of four colors (or four levels of gray scale). The entire picture is stored in a 520192-bit 600-ns buffer memory; it is read out at the rate of 10 pixels per microsecond to produce the video signal for the TV monitor.

Display Screen Considerations. The advantages of the TV scan display used for this work include the ability to display several 100000 pixels without flicker, to change any one pixel within a few microseconds without affecting the rest of the display, and to control the color of individual pixels for attracting attention to or otherwise label individual parts of the display. An alternative method for rapidly displaying the considerable volume of pixels needed for this application is the "storage screen" used for the Tektronix 4014 computer terminal. No display buffer is needed, because the storage screen provides its own picture memory. Therefore, even more picture data can be displayed than on the TV screen, because refresh rate and memory size limitations do not apply. However, the storage screen has no gray scale or color, and the entire screen has to be erased in order to erase even a single pixel.

(*v*) A 240 lines/inch scanner (Dest Data), which scans a full 8×10-inch page in 5 seconds (it was slowed to 12 seconds because our original disk drives were not fast enough to record the scan data). A fast scanner is an essential part of the interactive terminal; it allows convenient revision of artwork by using pen on paper (the method artists are familiar with), followed by rapid rescanning. The required scan resolution is about 700 lines/inch. Scanners of this resolution cost well over $100,000, however, which was our reason for selecting a less expensive 240-lines/inch device. The necessary resolution can still be achieved by scanning pictures which are three times larger than the size to be printed. An additional reason for selecting this scanner was that it uses a line of 2048 rigidly assembled photosensitive elements to digitize individual scan lines; it thus eliminates the need for rapidly moving parts, particularly the sensitive scanning mirrors. It has indeed proved itself to be a very reliable instrument.

(*vi*) An electrostatic printer (Versatec) which has facsimile quality

of 200 lines/inch. This printer provides immediate copies of designed advertisements, although of lower quality than the typesetter; it could be located remotely and connected by telephone line to the terminal (or to any central store of digitally stored advertisements or of "speculative art"). The picture in Fig. 3 was outputted on the electrostatic printer: its quality can be compared with that of the typesetter output shown in the abstract.

V. NOVA SOFTWARE

The terminal runs under MRDOS, Data General's Mapped Real-Time Disk Operating System, but user familiarity with MRDOS is not required. Most of the software was programmed in Data General Fortran V. Some bit and pixel operations were coded in assembler to increase execution speed. Also in assembler are the I/O routines for such nonstandard MRDOS devices as the tablet, the scanner, the TV display, and the Versatec. All I/O to disk and magnetic tape is managed by calls to the appropriate MRDOS subroutines. A number of load-on-call overlays are employed to accommodate the terminal programs in the available core space.

VI. USER ENVIRONMENT

The principal way the user communicates with the terminal is by operating the keyboard and the tablet/stylus. The layout is shown on the TV screen and immediately indicates any changes made by the user. The display colors are white, red, and blue on black background.

Man-Machine Communication Considerations. Experience has shown that considerable attention must be paid to the design of a command language to make control of the terminal easy for noncomputer people. The first terminal[4, 5] used picture control commands that were close relatives of the UNIX*[6] text editor commands. They consisted of abbreviated 1- and 2-letter command names typed on the keyboard. Text editing was also available and was done by context search; use of the stylus was limited to positioning operations.

Discussions with advertisement make-up experts indicated the need for an easier-to-operate command structure. Light buttons were considered, but would have further reduced the already small display area. Therefore, a command structure utilizing a function keyboard was chosen, similar to one we have already used successfully[7] in a speech signal manipulating program.

Each command consists of two parts: function and argument. Each part is specified by one key on the keyboard. A template laid over the keyboard relabels the keys for that purpose (Fig. 2). The template offers the following advantages: Labels on the template allow more meaningful names than 1- or 2-letter abbreviations, functions and

---

* Trademark of Bell Laboratories.

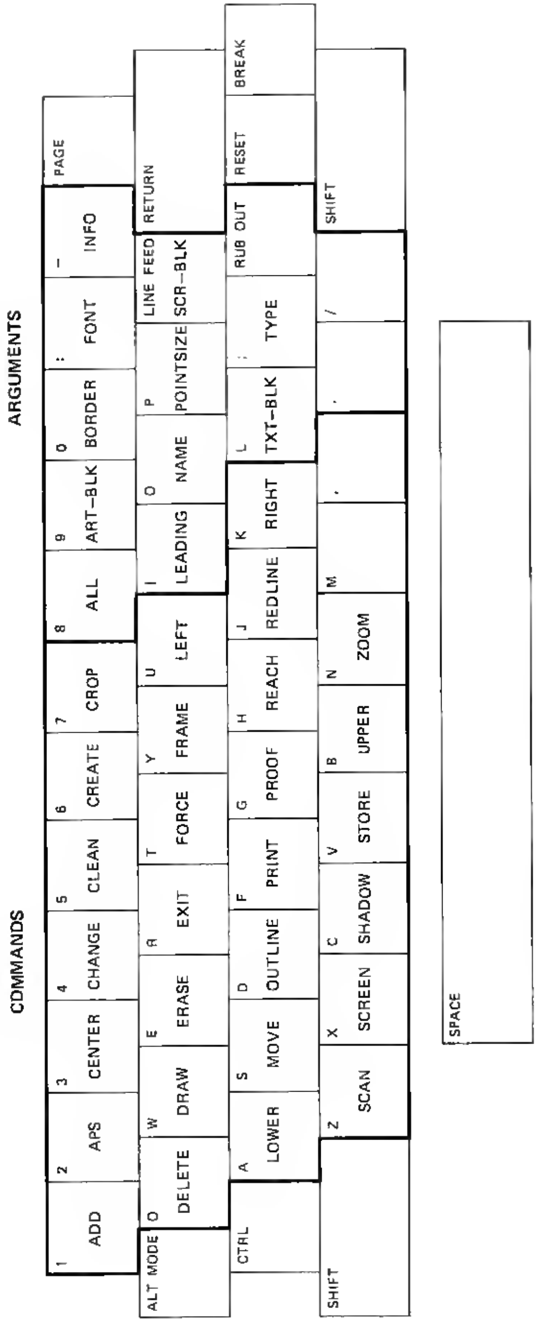

Fig. 2—Arrangement of commands and of their arguments on the keyboard.

arguments can be grouped separately, irrespective of their names, and the keyboard can still serve as a regular text input device.

Most commands are completed with the stylus by pointing to the additional information required. For example, text editing is performed by pointing directly to the character string to be changed followed by the replacement text from the keyboard. The terminal always mirrors the stylus position on the tablet as a tracking cross (cursor) on the TV screen. The cross consists of a thin vertical and horizontal line extending across the full screen. The data for the cursor lines are exclusively ORed with the display buffer. Thus, the lines remain always visible, though their color changes from the normal white-over-black background to black over white, red over blue, etc; this change of color is very helpful in the alignment of blocks. The desired cursor position can be signaled to the terminal by pressing on the stylus and thereby activating a pushbutton on its tip.

To get the best possible display resolution, the program scales each advertisement so that its longer boundary fits the full TV screen. In addition, a ZOOM command is available to enlarge any smaller area within the advertisement to the full size of the screen. All commands remain available to the zoomed display.

Color is used in various ways to guide the user:

(*i*) Colored template labels help to identify function and argument groups.

(*ii*) Red text on the screen warns that the particular text line exceeded its allotted space.

(*iii*) Solid blue areas indicate screened regions.

(*iv*) A blue rectangle shows the initial, a red rectangle the new location or size under stylus control for blocks being moved or resized.

(*v*) All the blocks of the applicable type are outlined in white, whenever a block needs to be identified.

More complex commands utilize a menu approach, where all possible options are listed for the user, who then selects one from the keyboard. All but the "quit this command" option terminate back to the menu.

The bell on the keyboard alerts the user to events such as illegal command terminations, completion of various stages in the execution of a lengthy command (e.g., drop-shadowing), or request of a user response to a command's option menu.

An abort pushbutton situated on the right-hand side of the keyboard lets the user abandon execution of the current command.

VII. GRAPHICS SOFTWARE

7.1 Blocks

The terminal uses layout blocks similar to those used in current production methods for the composition and paste-up of pieces of text

and artwork. Blocks are the key feature of the terminal, since all commands operate on the blocks themselves (e.g., MOVE) or on the information contained within them (e.g., CHANGE TYPE).

Blocks are rectangular areas which can contain either text or artwork. Their size and position are shown by red outlines during the layout process. Blocks can be created, deleted, modified, and moved. It is permissible for them to overlap. In addition, the artwork contained in a block may be replaced, recropped, and/or edited. Type within a block may be edited, centered, left/right justified and its font, point size, or leading modified. Also, blocks may be framed (e.g., a box placed around a telephone number) with the thickness of the frame under user control.

Each block is allocated 34 bytes (maximum of 30 blocks per ad). The information stored for each block includes its size, position within the ad space, whether it is to be framed, and the thickness of the frame. For art blocks, it also includes the artwork's disk file name and its cropping parameters, which determine what part of the artwork will actually appear within the block. The information for screen blocks is identical to that of art blocks, except for the additional screen density parameter. The text block information includes a justification parameter and pointers to the block's first and last text line in the text buffer.

7.2 Fonts

A font is generally defined as the complete assortment of characters of one size and style needed for ordinary composition. Each font character is digitally represented as a series of points within a rectangular area, called the character matrix. Different character sizes are plotted by enlarging or reducing the matrix accordingly.

To generate a character on the phototypesetter, the typeface, point size, and character identity must be specified. This information is used at composition time to retrieve a coded version of the character matrix from the typesetter's font library and to output the font shape using the typesetter's own special decoding circuitry. The terminal's software also includes dot matrices for the characters of several fonts; they are used to produce the character shapes on the Versatec and the TV screen. The characters are plotted point by point, and they are sized by horizontal and vertical scaling of the dot matrix.

The experimental terminal's font library currently contains 16 frequently used Yellow Pages fonts. A separate program prepares and installs each font into the library of the terminal. Additional fonts can be added at any time. The resolution of the installed typefaces is equivalent to a 48-point master on the typesetter. This is also the largest permissible character size in a display ad. The accuracy was

needed for the production of high-quality outlined and drop-shadowed type, which is generated by the terminal and plotted as art on the typesetter.

The fonts are stored in one disk file with a record format of 512 bytes/record. The first record is an index to the font data in the file. Each index entry consists of three 2-byte parameters: (*i*) the font number, (*ii*) a pointer to the font's first data record within the file, and (*iii*) the font's master sizes available on the typesetter. Each font can have a maximum of 128 characters.

For each font, the character data are preceded by four records, each containing two 256-byte tables. The tables contain such entries as the width and height of each character matrix, the left and right side bearing values, the amount of clearance above each character, the starting records of the character matrices, and the number of data records per character. The data are run-length encoded. Each character starts at a record boundary.

7.3 Text

There are two ways to enter text into the layout: Lines are typed in from the keyboard or text is read in from a separately prepared text file. Advance text preparation is more suited for the production of new ads, while keyboard input seems more advantageous for updating existing ads.

Three parameters control the appearance of type in one text line. (*i*) The font number specifies the typeface, (*ii*) the point size defines the size of the characters, and (*iii*) the leading parameter determines the extra spacing between the characters of consecutive text lines. Leading is executed before the characters are plotted and can possess negative values.

All text information is stored in a separate array (currently, 2000 bytes). Text lines, together with their font, point size, and leading parameters, are stored in the array upward. Pointers to the text lines run from the top of the array down. Each pointer consists of two entries, an array subscript to the start of each line and the line's character count.

Text lines are always linked to a text block. Deletion of a block results in the removal of its type as well. Type cannot be moved across text blocks.

Text commands perform substitution, insertion, deletion, and upper/lowercase conversion. Font, point size, and leading parameters may be altered separately or combined into a single command. The above commands can be applied to any string of consecutive characters within a text line, or to a sequence of entire text lines, or to a whole text block. Text in a block can be centered, left-justified or right-

justified. Special effects such as outlining or drop-shadowing may be applied to individual text lines. The outlined or shadowed text can be screened.

7.4 Artwork

The terminal handles two-tone (black-and-white) pictures such as ink drawings, logos, trademarks, and other illustrations which are scanned and stored digitally on disk files. Logos and artwork which might be needed for more than one advertisement are kept under a separate file directory and are accessible to all advertisements. Special artwork for individual ads is stored under the same directory as the layout specifications.

All artwork is digitized on the terminal. The difference between typesetter and scanner resolutions (3:1) can be compensated for by enlarging the input picture to three times the intended output size. Each scan results in data from the entire 8 × 10-inch scanning surface. Thus, one or more pictures may be digitized together, as long as they fit the scanning surface. The inputted data are immediately displayed on the TV screen, where individual pictures can be isolated and saved on disk. The cropping boundaries are defined by using the stylus/tablet to position vertical and horizontal lines.

Art data are currently stored without compression because the 96 Mbyte disk should have sufficient capacity to store picture data for the time-and-motion trial now in progress. However, if needed, average compression of about 5:1 would not be difficult to implement by applying the run-length coding programs we already use for storing font data. The pel data are stored on disk, scan line after scan line. A dummy scan line at the beginning of the file is used for storing scan parameters. The number of 16-bit words in each scan line is stored in the first 16-bit field of the dummy line; the total number of picture scan lines in the file is stored in the second 16-bit field.

Artwork is first scaled to fill the art block, often resulting in different horizontal and vertical scale factors. The FORCE command adjusts either the width or the height of the art block, so that the ratio is the same as that of the original art. The side to be changed is under user control.

Artwork may be screened, outlined, or drop-shadowed. A pencil/eraser function is also available for minor touchups. More extensive corrections should be made on the original, which should then be rescanned.

7.5 Special functions

In Yellow Pages production, the most frequently used artistic effects that involve time-consuming photographic techniques are screening,

outlining, and drop-shadowing, shown in Fig. 3. Algorithms to automate these processes were developed by Franklin[8] as part of an interactive picture manipulating system. The algorithms were then adapted and programmed into suitable commands for the terminal.

Outlining and drop-shadowing are normally applied to text only, to create custom lettering from conventional fonts. The commands, however, are completely general and can be used for interesting effects on artwork also. Representative specifications of thickness of outline and size and direction of shadow were selected in advance and are not

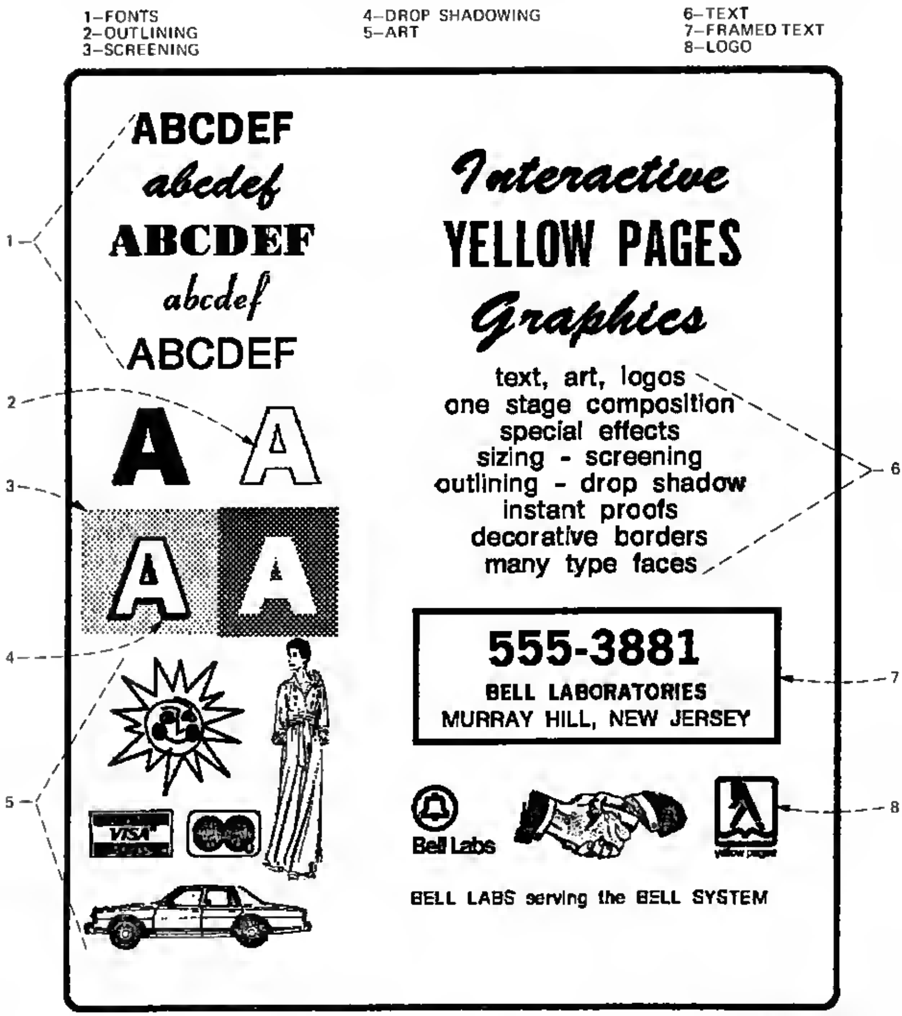

Fig. 3—Sample advertisement showing most of the terminal's capabilities. This picture was outputted on the electrostatic printer. Its quality can be compared with that of the typesetter output shown on page 2189.

under user control. The specifications are:

Width of outline:	10 decipoints (= ⅒th point)
Width of shadow:	30 decipoints
Position of sun:	Northeast.

The specifications are for 48-point objects; the thicknesses vary proportionately for other sizes.

7.5.1 Outlining

This operation converts a black-and-white (input) picture into a new (output) picture in which only the black-and-white transitions of the input are shown.

For this purpose a 3×3 window is shifted bit by bit across the entire input picture. The center element of every matrix position is modified according to the status of its surrounding elements and transferred to the corresponding position of the output picture. If all surrounding elements are zero, the center element is transferred as zero (no threshold). If any surrounding element is nonzero, the center element is transferred complemented (possible threshold).

The generated outline in this case is always just one pixel wide. Larger windows would generate thicker outlines but would require a greatly increased number of operations. The terminal, therefore, always uses a 3×3 window and generates the required thickness on the output picture by plotting a series of points for every threshold value found.

In the process of outlining text, the terminal first converts a text block into an art block and a text line into a digital picture file. The font data used for conversion come from the terminal's local font library.

7.5.2 Drop-shadowing

This operation generates a new (output) picture consisting of the outlined (input) picture with a shadow added. Initially, we again intended to use an algorithm by Franklin. However, this was changed soon to a different method,[9] which required less core space and avoided the processing of individual bits.

Franklin's algorithm computes the shadow by shifting a matrix bit by bit across the entire input picture. The size of the matrix determines the length of the shadow. The corner element of the matrix selected to compute the corresponding output element determines the direction of the shadow. An output element is zero if the input corner element is nonzero or if the corner element and all the other elements along its diagonal are zero. An output element is nonzero if the corner element

is zero and any of the other elements along the diagonal is nonzero. To achieve the drop-shadowed effect, the outlined picture and the shadow are ORed together.

The new algorithm combines an input scan line, the corresponding output scan line and as many subsequent output lines as are needed for the length of the shadow; initially, all output scan lines are zeroed. The shadow is calculated by first ORing the scan lines of the initial and the outlined picture together. These data are ORed to the corresponding output scan line. The data are then shifted left by one bit and ORed to the following output (shadow) scan line. This operation is repeated, always shifting the input data one more bit until all the shadow scan lines are processed. Finally, to mask out any hidden shadows, the current output scan line is exclusively ORed with the initial picture scan line and written on the output file. Before processing the next input line, the first shadow scan line becomes the current output scan line, the second shadow scan line becomes the first one, and so on; the last line is zeroed.

7.5.3 Screening

In this operation, the computer program traces the boundaries of any area enclosed by a continuous line and then fills this area with screening dots of selected density. The area to be screened is identified on the TV screen by pointing to any part of it with the stylus. The screened area is recorded on a separate screen file.

On the final output, screened areas are shown as regions filled with evenly spaced spots of uniform size. The chosen spot size determines the perceived gray level (screen density). Areas of different density are recorded in separate files. If no screen file exists at the start of the screening operation, the file is created and zeroed.

The tracing algorithm employs a stack to manage the starting points for the boundary search. The first stack entry is the initial starting point. When the stack becomes empty the search is complete. For each starting point, the scan line to the left and right of the initial position is searched for a zero to nonzero boundary. The elements between the left and right boundaries are then set to nonzero on the corresponding screen line. Finally the immediate scan line above and below is searched between the two boundaries from left to right for nonzero to zero transitions. (A nonzero element is assumed beyond the left boundary.) The transition point (zero) coordinates are pushed on the stack. This procedure is repeated for all points on the stack. The scan lines being searched are the picture scan lines ORed with their corresponding screen lines. To improve efficiency and minimize the required I/O, the stack is first searched for another point on the current scan line. If

found, this point is popped and it becomes the next starting point; if the search fails, the last point entered will be popped from the stack.

7.5.4 Pencil/eraser

This function uses the tablet/stylus to simulate a pencil/eraser. It affects the artwork on the TV screen and, simultaneously, the data on disk. The function is best suited to correct blemishes and to straighten out lines. It was not intended for freehand drawing or curves. Within its scope, it has proved to be convenient and easy to use.

At the start, the user selects a particular section of the picture as the work area. For this purpose, the program first shows the complete picture on the screen. The chosen work area is then redrawn on the full screen and the pencil/eraser "tip" appears on the screen.

The tip is shown as a white (pencil) or red (eraser) rectangular area and follows the movement of the stylus. Actual drawing/erasing occurs only while the pen switch is depressed, with the mode depending on the status of the draw/erase sense switch. In erase mode, the pels covered by the tip are set to zero, in draw mode, to nonzero. The size of the tip is adjustable. Adjustment is initiated by the TIP pushbutton at the right-hand side of the keyboard, which cancels the current tip and starts the cursor for creating a new one. The task to be accomplished should influence the shape of the new tip. For example, a small square tip is useful for fine detail and isolated blemishes, a tip in the shape of a narrow horizontal or vertical bar is more suited to smooth out ragged edges along horizontal or vertical lines.

The user can specify a scale factor, which controls how large each picture element will appear on the screen. Larger scale factors are more suited for work with intricate detail, because the tip is less sensitive.

User requests, other than a tip size change, are initiated by the ABORT pushbutton, which terminates the draw/erase loop and causes return to the options selection loop.

The pencil/eraser function is available both as a command and as a menu item under the SCAN command.

VIII. COMMAND SUMMARY

Command names are listed below in alphabetical order; a list of their possible arguments is included in square brackets following each command name. A more detailed description of the commands can be found in the appendix.

ADD	[border, name, type]
APS	[all]

CENTER	[all, type, text-block]
CHANGE	[border, name, type, font, point size, leading, text-block, art-block, screen-block]
CLEAN	[all, art-block, screen-block]
CREATE	[art-block, text-block]
CROP	[art-block, screen-block]
DELETE	[all, name, type, text-block, art-block, screen-block]
DRAW	[all, text-block, art-block, screen-block]
ERASE	[all, text-block, art-block, screen-block]
EXIT	[all]
FORCE	[art-block, screen-block, point size, leading, font]
FRAME	[text-block, art-block, screen-block]
LEFT	[all, type, text-block]
LOWER	[all, type, text-block]
MOVE	[text-block, art-block, screen-block]
OUTLINE	[text-block, art-block]
PRINT	[info, all, border, type, text-block, art-block, screen-block]
PROOF	[all]
REACH	[text-block, art-block, screen-block]
REDLINE	[all]
RIGHT	[all, type, text-block]
SCAN	[all]
SCREEN	[art-block]
SHADOW	[text-block, art-block]
STORE	[all]
UPPER	[all, type, text-block]
ZOOM	[all, art-block]

Argument names are self-explanatory, except for ALL. In some cases, ALL implies "everything" as in ERASE ALL (blocks); in some cases ALL implies "everything of a certain kind" as in LOWER ALL (type); in some cases it is simply a dummy argument to conform to the standard command sequence as in SCAN ALL.

9. OUTPUTTING TO THE TYPESETTER

9.1 Overview

The final task of the terminal is to output the entire advertisement (text and pictures) on a typesetter; this typesetter output is then used for making the printing plate. Our terminal computes commands for the Autologic APS-4 typesetter which are then recorded on magnetic tape. The tape is later mounted on the typesetter and the advertisement is outputted without further attention.

The APS-4 was used for our experiments because it was the typesetter available where the terminal is being field-tested. Later the more modern APS-5 was made available to us and our research profited by

being allowed to use it. The knowledge gained with the APS-4 and APS-5 should be easily transferable to any other CRT (or laser) typesetter.

Considerable study, testing, and typesetter adjustments were needed before the typesetter would produce pictures with the quality and speed needed for directory production. This was so even though the APS-4 (and other CRT typesetters now in use at telephone companies) already has basic picture-drawing capabilities which are used for drawing font shapes when outputting text. However, these font drawing methods could not conveniently be adapted to drawing more complex images. Another way available for drawing pictures was to adapt the APS's "rule" commands, even though they were originally intended to draw single straight lines only. We found that considerable testing (by ourselves), tuning, and modification (by the manufacturer) was needed to make this method usable; even then, its quality was not entirely satisfactory and its speed slower than desirable. Finally, a new method was introduced for graphics output on the APS-5 (called the "graphics" mode), and we were its first users.

As a result of these experiments, we have demonstrated a reliable method for typesetter graphics which produces pictures whose quality, for directory purposes, compared well with those produced by photography. Also, speed trials using about a half-dozen four-column pages showed that entire pages filled with representative graphics material and text were produced in about 90 seconds, which was considered a very satisfactory performance.

9.2 Graphics on the APS-4 and APS-5

9.2.1 The nibble code

The instruction set of the APS-4 and APS-5 includes the usual beam positioning commands and specifications for font, point size, and leading. Text characters are painted by closely spaced vertical strokes of the CRT beam. The font shapes are specified in terms of the end points of the vertical strokes. The stroking data are coded into a form of run-length-differential code. Individual codes consist of 2-bit units, which is the reason for calling them "nibble" code (by contrast to the 8-bit byte). The APS-4 and APS-5 use a hardwired nibble decoder whose output controls the CRT beam. This decoder can only deal with a maximum of four vertical strokes in one "column" and is therefore not convenient or efficient for more complex shapes.

9.2.2 The rule command

The APS machines also have "rule" commands which generate horizontal lines of selectable width and height. The lines are again produced by a series of closely spaced vertical strokes. This is the first of the two different methods we used for typesetting graphics. Its principal drawback is that fractional data about rule lengths are not

handled adequately (rounding troubles), so that the end points of a number of chained "rules" along a single horizontal line quickly become inaccurate; this results in a ragged appearance of right-hand outlines of complex pictures. Most of these troubles were remedied by Autologic after we demonstrated shortcomings to them. However, in pictures that consist of large numbers of short "rules" (such as a screen pattern), the trouble has not completely disappeared. The method is also relatively slow.

Pictures to be typeset using rule commands are drawn by plotting individual scan lines. For each line, the picture elements are analyzed for zero and nonzero run-lengths. Nonzero sequences are used to set rule widths, and zero sequences are translated into beam positioning commands. Any required scaling is performed by direct sampling of the input data. The height of rules is normally set to one decipoint. The height of the rule is increased only when the picture has to be enlarged. In this way, the rule has to be drawn only once, instead of plotting the data several times as required by the ratio of enlargement.

Screen data are typeset similarly, except that nonzero sequences are translated into dot sequences instead of solid black rules. Traditionally, screen measures are expressed in diagonal lines per square inch, with screen dots arranged along parallel lines angled at 45 and 135 degrees to a horizontal line. On the typesetter, however, we plot *horizontal* screen lines. Each dot is drawn as a single square rule; its size is determined by the screen density. The dot spacing (in decipoints) along horizontal lines for a given typesetter can be computed to

$$s = sqrt(2)*r/d$$

where r is the resolution of the typesetter (in decipoints/inch) and d is the diagonal screen measure (in screen lines/inch). Vertically, screen rows are repeated in intervals of half the horizontal spacing, with alternate rows being staggered by the same amount. For example, an 85-line screen on the APS-4 with 720 decipoints per inch resolution has a horizontal spacing of 12 decipoints. Consecutive screen rows are 6 decipoints apart, with every other row indented by 6 decipoints.

Before plotting a screened data file, the program initializes the width and height of a screen rule. Also, a basic command loop is set up in one of the APS buffers: Draw one rule, move beam horizontally to start of next rule, decrement loop count, quit loop if count is zero, otherwise repeat loop. Subsequently, screen sequences scan can be plotted by just two APS commands: Set loop count to the number of screen dots required and execute the command string.

9.2.3 The graphics mode

Recently, Autologic developed an alternative graphics feature on the APS-5 for faster and more flexible typesetting. Again, vertical

stroking under the control of the hardwired nibble decoder is utilized. The method is based on a special font with 256 characters identified by an 8-bit code. Each character consists of up to four vertical strokes in a single column. The pattern of strokes in each special character is the same as the pattern of ones in the 8-bit code which specifies it. Pictures are typeset in a swath of eight horizontal scan lines at a time; 8-bit character codes for specifying a picture are derived by slicing the eight scan lines vertically at every pixel position. The "graphics" mode provides a repeat (run-length) code to avoid repeating identical codes. Thus fewer bits are needed to specify a picture, resulting in less storage and higher output speeds.

For screen data the scan lines are logically ANDed with the screen pattern before remapping; otherwise, the process is the same as for artwork.

One item that has not yet been resolved concerns lens magnification. For text, the typesetter program compensates for the different lenses, and the resulting output always has the same size. For the "graphics" mode, no lens compensation is performed. Artwork with a scale factor of 1.0 will be properly sized only on a machine with a 2.22X lens. To obtain the same result on a machine with a 2.66X lens, the user must adjust the artwork's scaling parameters to 0.8326.

9.3 Results

Of major interest in typesetting graphics are output speed and quality.

To obtain a meaningful timing comparison, we produced the same page of 30 logos (Fig. 4) by the "rule" mode on the APS-4 and on the APS-5 and by the new "graphics" mode on the APS-5. The logos had been scanned previously and were outputted in six rows on a single page using the rule and the graphic modes.

The typesetter timing tests for the entire page of 30 logos gave the following results:

(*i*) Using rule commands the APS-4 was over four times faster than the APS-5.

(*ii*) Using the graphics mode (on the APS-5), was seventeen times faster than the rule commands on the APS-5 and four times faster than the rule commands on the APS-4.

The output quality of artwork (except for screens) produced by either the rule or the graphics methods was very high. The graphics mode also produced excellent, uniform-looking screen patterns. Occasionally, a series of dots in a column appeared fainter but were still uniformly spaced. This effect can be explained by the grain of the paper and varies with screen measure and density. Screen patterns produced by rule commands on the APS-5 showed regularly spaced

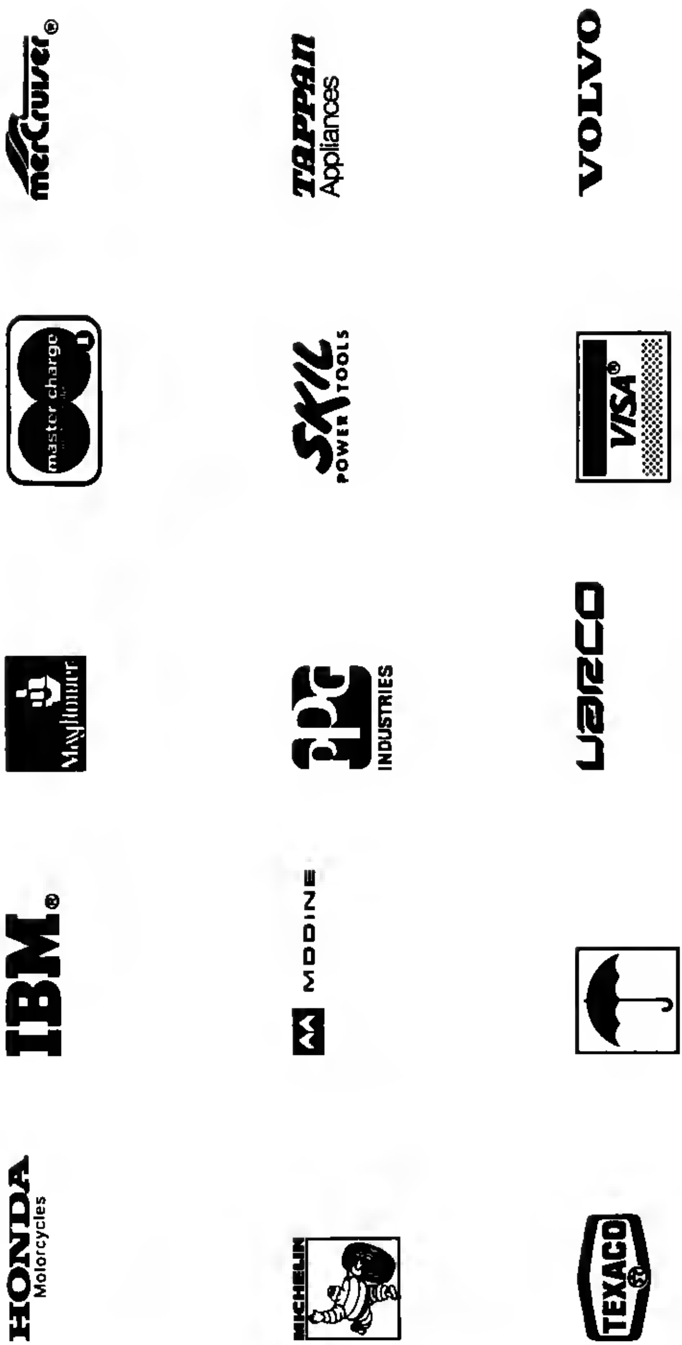

Fig. 4—Thirty logos used for timing.

thin vertical stripes. They seem to be introduced by the APS-5 when mapping absolute decipoint values to actual beam distances, at which point the lens magnification also has to be considered. Various attempts to remedy this problem did not entirely succeed. Results on the APS-4 varied with the particular machine used.

X. FIELD TRIALS

Two different field trials of the experimental terminal have been or are being carried out. The goal of the first trial—which has already been completed—was to establish what type of personnel is best suited to operate the terminal, to gain experience for establishing operating routines, and to identify modifications for better operation, particularly for making the operator's task simpler. The goal of the second trial—now getting under way—is to make cost and time comparisons between the new terminal and the conventional methods of producing advertisements.

In the first trial, two persons were trained in the use of the terminal. One had years of artistic layout experience, the other trained clerical personnel for various tasks. After four weeks, a series of program changes were installed to improve the reliability of the terminal, to increase feedback to users, and to adopt certain terminologies more meaningful to the users. The following changes were made:

(*i*) The command key sequence was standardized to two keys followed by a carriage return.

(*ii*) The capability of scanning partial pages was removed; it was more efficient always to scan the whole page and crop the material on the TV screen.

(*iii*) Some new messages, such as "scan in progress," were added to inform and reassure the user on the status of commands which required more than 1 or 2 seconds for execution.

(*iv*) Some terminology was changed to names more meaningful to users: pictures became "art," text lines and text strings became "type," output from the electrostatic printer was renamed "proof."

(*v*) Automatic saving on disk of the layout specifications after every command was added to facilitate painless continuation after an unexpected interruption or program abort. After these changes, evaluation of the terminal proceeded smoothly.

The lessons learned from the first trial were as follows:

(*i*) The terminal could indeed be a valuable tool.

(*ii*) Clerical personnel can operate the terminal, especially if the advertisement worksheet—scanned and displayed on the TV screen—could be used as a guide for positioning and sizing the layout blocks.

(*iii*) The terminal should be used mainly for the input and layout

of artwork; text should be prepared separately in advance and only modified online.

(*iv*) A pencil/eraser option must exist to edit the scanned data, such as to delete erroneous spots and possibly touch up logos and artwork.

(*v*) There should be a second trial to make a more detailed comparison (including a time and motion study) between our terminal and conventional production methods; a large number of ads (100 or more) would have to be prepared by both methods.

(*vi*) A clerk must be trained and become proficient in the use of the terminal before the study can begin.

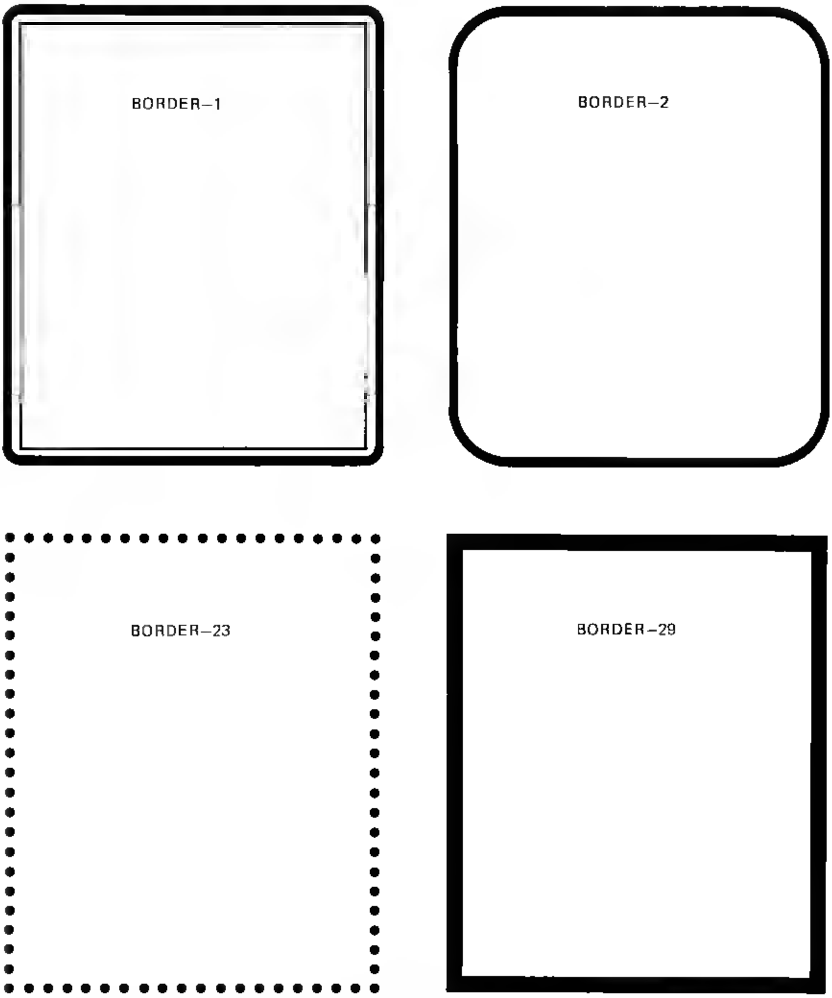

Fig. 5—Four of the ten decorative borders available on the terminal.

(*vii*) To produce real ads, the terminal must have a reasonable number of different fonts, decorative borders (Fig. 5), and adequate storage facilities for approximately 100 illustrations, logos, and stock art.

The second field trial has just started. Its objective is to collect data for a cost and time comparison between the current advertisement production method and the experimental terminal method. To establish costs, the layout process is subdivided into individual steps (such as scanning of worksheet, scanning of artwork, adding a screen) and each step is to be timed separately. A set of 50 advertisements was selected which covered the variety of material considered typical of those appearing in Yellow Pages directories. Statistics on the type of revision work normally encountered were also collected. In the first part of the trial, the 50 advertisements will be created afresh and timed on the experimental terminal, just as if they were new advertisements. Later, the same time measurements will be made for "revising" existing advertisements, using the 50 advertisements prepared in the first part of the trial. Separate estimates will be made of the economic and other advantages that result from greatly shortening the time span between closing a directory and its printing. Further, the results from this field trial should help in the assessment of future terminal needs and define areas for further study.

The terminal had to be expanded considerably to accommodate the much greater variety of work and the simulated production environment. Preparations for the trial included:

(*i*) The installation of a 96 Mbyte disk for storing the additional fonts, illustrations, and logos.

(*ii*) The installation of 15 additional typefaces and the introduction of run-length coding for storing them in less disk space.

(*iii*) Programs to generate 10 stock borders by rule.

(*iv*) Pencil/eraser capabilities.

(*v*) A program for separate text input in advance.

(*vi*) The capability of framing text, such as placing a box around a telephone number.

(*vii*) Selection and training of a clerk in the use of the terminal.

All preparations have now been completed and the trial is under way.

XI. CONCLUSIONS

The work described in this paper demonstrates that modern directory production methods are able to process pictures with the same ease with which they currently handle text. In our experimental terminal, photographic methods were replaced by digital methods to position, size, crop, screen, outline, shadow, and store pictures conveniently and efficiently; methods for typesetting artwork with high speed

and quality have also been established. As a result, directory pages can now be typeset in a single pass, without paste-ups, and corresponding production cost reductions and sales advantages can be expected.

The new techniques demonstrated on our experimental terminal are to be included in the full production prototype for the preparation, maintenance, and conversion of all Yellow Pages art material which is now being developed by New York Telephone Company with AT&T support.

XII. ACKNOWLEDGMENTS

We would like to express our thanks for the imaginative, forward-looking, and substantial help and encouragement we received from Bell System companies. In particular, we are grateful to P. J. Desmond and B. R. Jansson (New York Telephone), M. Jensen (Northwestern Bell), J. C. O'Neel (Pacific Telephone), and E. R. Buckstine (AT&T). B. R. Moore, Marlene D. Bierig, and J. A. Zanette (all at Pacific Telephone) contributed expert advice and hard work in helping us improve the all-important man-machine communication aspects of the terminal and in carrying out the field trials. We thank R. J. Cain and H. H. Finney (both at New York Telephone) for their help in making the APS-4 and APS-5 produce acceptable graphics. We also wish to acknowledge gratefully the contribution of O. C. Jensen of Bell Laboratories, who designed, constructed, and maintained the terminal's graphics hardware.

APPENDIX

Command Description

The functions of the various commands are briefly explained here. Multipurpose commands appear under each appropriate subheading. Detailed operating instructions can be found in the user's manual.[10]

A.1 General block commands

CHANGE is used to adjust the dimensions of a specified block within the work space.

CREATE generates a new block of the specified type.

DELETE deletes the indicated block and its associated information from the layout.

DRAW, ERASE operate on the TV display only. DRAW displays the selected block on the screen, ERASE erases it. The block will still appear on the final output.

FRAME lets the user specify an outline or "frame" around a block for the final output. The thickness of the outline is specified in ⅒ point units.

MOVE repositions the specified block within the layout space.

PRINT prints the coordinates (relative to the top left corner of the advertisement) of the designated block and, for art or screen blocks, the associated data file name and, for text blocks, the text lines together with their font, leading, and point size parameters.

REACH allows the user to address a block completely contained within another block of the same type.

A.2 Text handling commands

ADD allows the user to add type to an empty block or to insert or append type to existing text lines in a block, or it can get the data for all text blocks from a previously prepared text file.

LEFT, CENTER, RIGHT are used to justify or to center text lines in the specified text block.

CHANGE is used to edit character strings or text lines, or modify font, leading, and/or point-size parameters.

DELETE is used to delete character strings or text lines.

UPPER, LOWER convert character strings, text lines, or text blocks to upper- or lowercase. LOWER is the more frequently used command, since all input from the Tektronix keyboard is initially in uppercase.

FORCE sets the text parameters of the specified block back to their default values. This command is useful after a change of the text parameters causes type to exceed the allocated block space.

PRINT is used to get a listing of the indicated text lines with their current font, leading, and point-size values.

A.3 Art processing commands

ADD, CHANGE are used to associate a picture data file with a specified art block.

CLEAN invokes the pencil/eraser function to touch up the artwork in the current art block.

CROP is used to frame the original artwork. Only the data inside the frame will appear in its associated art block. This feature allows the same artwork to be used for various ads, with each ad showing different parts of the stored artwork.

DELETE deletes irretrievably the specified picture data file from disk.

FORCE adjusts the parameters of art or screen blocks in such a way that the associated picture has the same scale factor in both the horizontal and vertical dimension, and thus appears undistorted. This is important for trademarks and logos. For other artwork, the layout artist may actually prefer the effect of having the artwork appear condensed or expanded along one dimension. FORCE allows the user to specify which of the four sides of the art block should remain unchanged in size as well as position within the workspace.

PRINT lists the file name associated with the data of an art or screen block.

SCAN is the command to convert artwork from a paper drawing into digital form for handling by the terminal. The command employs an options menu to perform various operations connected with scanning: crop the scanned data, store the data under a user assigned name, copy data from one file or device to another for backup or restoration purposes, print some file characteristics, and use the pencil/eraser.

A.4 Commands for producing special effects

OUTLINE produces a new picture data file that contains only the outlines of the initial artwork or the initial text characters.

SHADOW produces another picture file that contains the outlined picture with a shadow added.

SCREEN permits the user to specify areas for screening.

A.5 Commands for producing output on the typesetter or on the electrostatic printer

APS computes and records on magnetic tape the commands to drive the Autologic APS-4 or APS-5 phototypesetters to produce the final version of the ad.

PROOF calculates and then plots an image of the ad on the electrostatic printer.

A.6 Miscellaneous other commands

CHANGE is also used to assign or change the decorative border around the ad.

EXIT is the command to wrap up the layout program properly.

PRINT can also be used to get the overall statistics of the current ad. In this case, the size of the ad, the number of blocks of each type, the number of text lines, and the decorative border type are listed.

REDLINE reverses the current status of either showing or not showing the red outlines around all blocks during the layout process.

STORE is used to save the layout specifications under a user specified name. The information includes everything except the picture and screen data files.

ZOOM allows the user to enlarge a specific part of the ad across the full screen. All commands remain available to the zoomed ad.

REFERENCES

1. A. J. Frank. "High Fidelity Encoding of Two-Level, High Resolution Images," IEEE Inter. Conf. of Commun., June 1973, pp. 26-5—26-10.
2. A. J. Frank and R. H. Groff, "On Statistical Coding of Two-Tone Image Ensembles," Proc. SID, *17/2*, Second Quarter 1976, pp. 102-110.
3. R. G. Todd, "A Hardware Decoder for Two-Dimensionally Compressed Pictures," unpublished work.

4. I. G. Gershkoff, "An Interactive System for Page Layout Design," unpublished work.
5. P. B. Denes and I. G. Gershkoff, "An Interactive System for Page Layout Design," Proc. ACM Conf., Nov. 1974, pp. 212–221.
6. *UNIX*™ Programmer's Manual, Bell Laboratories, March 1977.
7. L. H. Nakatani, Computer-Aided Signal Handling for Speech Research," J. Acoust. Soc. Amer., *61*, (Apr 1977), pp 1056–1062.
8. D. L. Franklin, "An Interactive Picture Manipulation System," unpublished work.
9. K. C. Knowlton, Bell Laboratories, private communication.
10. B. E. Caspers, "User's Guide for the Yellow Pages Design Terminal," unpublished work.